Computers Math. Applic. Vol. 20, No. 7, pp. 1–10, 1990
Printed in Great Britain.

0097-4943/90 $3.00 + 0.00

A QUEUING NETWORK APPROACH TO THE TOPOLOGICAL OPTIMIZATION OF LINKED CLUSTER NETWORKS

Y. M. CHEN
Institute of Electrical and Computer Engineering, National Cheng Kung University, Tainan, Taiwan 70101, R.O.C.

L. M. TSENG
Institute of Information and Electronic Engineering, National Central University, Chung-Li, Taiwan 32034, R.O.C.

(*Received* 4 *May* 1989; *in revised form* 22 *September* 1989)

Abstract—Minimization on the queuing delay of messages is an important issue in the area of network topological design. This paper presents a queuing network approach, rather than conventional mathematical programming methods, to determine the connections of computing resources such that the average message sojourn time can be minimized. Using the analytic performance measure of the queing network as an evaluation criterion, we develop a heuristic algorithm to search for a nearly optimal topology. An empirical evidence for the success of this heuristic algorithm is given. By the heuristic algorithm, the performance measure of the nearly optimal topology that is found is normally within 5% of the measure of the optimal topology. The advantage of this approach is that we can properly model the queuing delay and directly relate the optimization criterion to the system performance.

1. INTRODUCTION

A linked cluster network is divided into a set of clusters, with each node belonging to one of the disjoint clusters. In each cluster there exists a cluster head acting as a local centralized controller for the nodes in its cluster. These kind of networks have received much attention, because they are applicable to a variety of networks, ranging from multiprocessors, local area networks, to satellite networks [1, 2].

A primary issue in designing such a network is to determine the appropriate set of nodes into a cluster [3, 4], i.e. to determine the optimal interconnection of computing resources in the network. This issue is typically solved by combinatorial algorithms, such as those presented in Refs [3–5], where the solutions of the backbone design and the local network access design are given. Though these solutions have considered various network factors, such as line capacities, communication costs, etc. they all suffer from the disadvantage that they did not properly model the queuing delays, thus the optimization criterion cannot be directly related to the system performance measure. On the other hand, as the network nodes may share some common resources (such as file server, communication controller, transmission lines, etc.) very frequently, the queuing delay may become significant and should be taken into account in the early network design phase [6–8]. In the following context, for example, we will show that in some cases, the differences of the average message response time between a randomly assigned network topology and the optimized network topology may be up to 60%.

Our approach in solving this optimization problem proceeds in two stages. In the first stage, we model the network as a product form queuing network [9, 10] to obtain an analytic performance measure. In the second stage, by taking this performance measure as an optimization criterion, we use a heuristic greedy algorithm to search for a nearly optimal topology.

For the search of an optimal topology, the complexity of the heuristic search is $O(N - C)$, while that of the exhaustive search is $O(C^{N-C})$, where N is the number of network nodes and C is the number of clusters. An empirical evidence for the success of this heuristic algorithm is given. The performance measure of the nearly optimal topology that is found by the heuristic is normally within 5% of the measure of the optimal topology.

In Section 2, we will briefly introduce an experimental network, named OCSE (office coordination supporting environment) [11], and describe its queuing network model (QNM). In Section 3, we solve this QNM and obtain the average message sojourn time as a system performance measure. Section 4 describes the heuristic search algorithm. The performance of this heuristic algorithm is also discussed in that section. Finally, in Section 5, we explore research directions and draw conclusions.

2. MODEL DESCRIPTION

2.1. System overview

The OCSE system is used for office automation (OA) and consists of many clusters which may be installed in an office or a department. Figure 1 illustrates the topological view of the clustered structure. Each cluster contains two kinds of stations: office workstations (OWSs), which support user interfaces and local processing (e.g. word processing), and office processors (OPs), which manage the electronic mailbox and distributed database for the information sharing among office workers. A user cannot directly access the OP, but may in turn send(receive) messages to(from) it through the OWS. All of the OPs and OWSs are connected to the specially designed communication units called switching service elements (SSEs), each of which acts as a cluster head in each cluster.

During message passing, the SSE puts the message into a dedicated buffer if the destination station is busy. Unlike the OWS and OP, the SSE neither interprets nor processes messages; it only establishes communication links for messages. Since all messages visit at least one SSE, the message buffer of an SSE can be treated as a large queue which has many job arrival sources. Similarly, every OWS and OP has a buffer to store incoming messages that cannot be processed immediately. Therefore, it is natural to model this OCSE system as a QNM. In the following, the parameters for a formal mathematical model are defined and some common assumptions are also made.

2.2. Mathematical model

In the sequel, capital letters denote sets while lower case letters denote elements of a set. In addition, we use the term average as a synonym for mean.

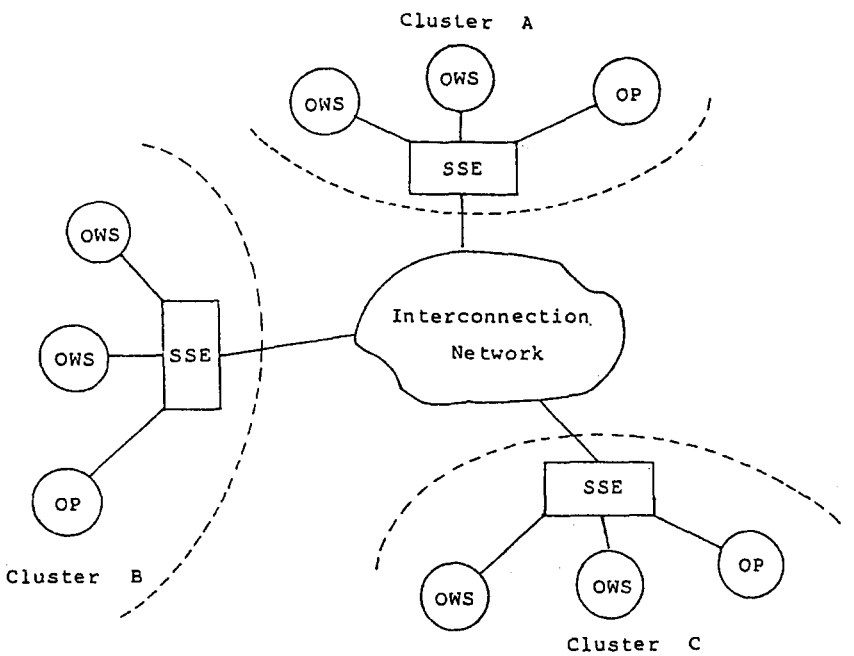

Fig. 1. Topological view of the OCSE system.

N—Total number of OWSs and OPs in the system.

N_k—The number of stations (including OWS and OP) in cluster k.

C—The number of clusters in the system. Since every cluster requires an SSE to control the communication links, C also equals the number of SSEs.

K—The number of service centers in the queuing network. In the current model, we assume that the communication medium is lightly loaded, so a message transmission requires an approximately constant average delay (W_d), which can be included in the processing time of the sending node. Therefore, K equals the sum of SSEs, OPs and OWSs. In addition, we also assume that each service center is scheduled by the FIFO discipline and has an exponential service time distribution of average rate $r_i, i = 1 \ldots K$.

(a_{0i}), where $i = 1 \ldots N$—An N vector, named EMA (exogenous message arrival rate) vector, that specifies the rate of message arrival from outside into each station. All of the exogenous arrivals are assumed to be Poisson processes. In the real system, these arrivals are generated either by the user of OWS or by the management software of OP.

$[P_{i,j}], i, j = 1 \ldots N$—An $N \times N$ matrix, named PIT (proportion of interstation traffic) matrix, denoting the message flow relations among stations, where

$$P_{i,j} = \frac{\text{the number of messages going from the station } i \text{ to the station } j}{\text{the total number of messages emitted by the station } i}.$$

It is noted that $P_{i,i}$ equals zero, for the messages emitted by a station must visit at least one SSE before return (if necessary) to the original station.

$L_{i,j} = (i, L^1_{i,j}, \ldots, L^m_{i,j}, j)$—An ordered set of successive clusters visited by messages sent from cluster i to cluster j. We assume that a message is routed along the shortest path between node i and node j. If two or more paths are of equal length, the first one is selected.

W—Average message sojourn time.

To simplify the expressions in the next section, we assume that all the messages in this system are of the same type, i.e. they have the same service demands in each service center. In addition, the queuing delays of every message in successive service centers are assumed to be nearly independent.

Since all messages will eventually receive their service and leave the system, this model belongs to the open QNM [10]. Moreover, we assume that all exogenous message arrivals are Poisson processes and all service time distributions are exponential. Therefore, this network model is a Jackson queuing network [9, 10]. The performance of such networks is easily obtained if we know the solution of individual service centers. Figure 2 shows such a queuing network example which contains two clusters.

3. QUEUING NETWORK MODEL SOLUTION

Though the values of the EMA vector and the PIT matrix can be estimated in system design phase or measured in system running phase, it is not easy to calculate the average message sojourn

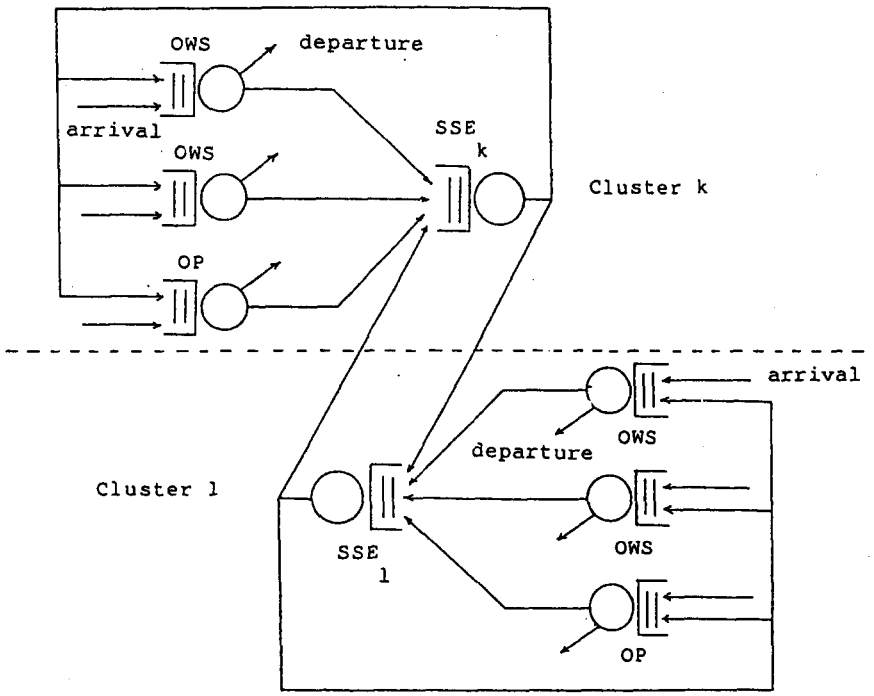

Fig. 2. A queuing network example.

time due to the queuing effects and the inclusion of SSEs in the system. In this section, we discuss the key issues in solving this QNM.

3.1. Calculation of service centers' message arrival rate

In the following expressions, $\bar{N}$ denotes the set of stations in the system, $\bar{C}$ denotes the set of clusters, and $\bar{N}_l$ denotes the set of stations belonging to cluster l.

Now, let a_i be the total message arrival rate, including messages from outside and from other service centers, into the service center i. Then given the EMA vector and PIT matrix, we can obtain

$$a_i = a_{0i} + \sum_{\substack{l \in \bar{C} \\ j \in \bar{N}_l}} a_{0,j} * P_{j,i}^{l,k}, \quad i \in \bar{N}, \quad k \in \bar{C}, \tag{1}$$

where

$$P_{j,i}^{l,k} = \frac{\text{the number of messages going from the station } j \text{ of cluster } l \text{ to the station } i \text{ of cluster } k}{\text{the number of messages emitted by the station } j \text{ of cluster } l}.$$

In the right-hand side of equation (1), the first term is the exogenous arrival rate, and the second term is the departure rates from other service centers. Since the departure process can be shown to be Poisson by the output theorem [10], and the superposition and decomposition of Poisson processes are still Poisson processes [12], it is easy to see that the total message arrival to service centre i remains Poisson process.

Similarly, the message arrival into the SSE of a cluster is Poisson with rate

$$a_s^k = \sum_{\substack{i \in \bar{N}_p \\ j \in \bar{N}_l}}^{p,l \in \bar{C}} a_{0i} * P_{i,j}^{p,l} - \sum_{\substack{l \in \bar{C}-k \\ m,n \in \bar{N}_l}} a_{0m} * P_{m,n}^{l,l} + \sum_{\substack{i \in \bar{N}_p \\ j \in \bar{N}_q}}^{k \in L_{p,q}} a_{0i} * P_{i,j}^{p,q}. \tag{2}$$

From equations (1) and (2), we can obtain all of the message arrival rates into service centers of the QNM.

3.2. Computation of average message sojourn time

By Jackson's Theorem [10], the distribution of the network state is completely decomposed into a product of individual service center's distribution, i.e.

$$\text{Prob}(n_1, \ldots, n_j, \ldots, n_K) = \prod_{j=1}^{k} \text{Prob}_j(n_j), \quad n_j \geqslant 0, \tag{3}$$

where n_j is the number of messages waiting or being served in node j, and K is the number of service centers.

The implication of equation (3) is that the mean network population and response time can be determined without ever dealing with the probabilities of network states. Instead, these performance measures are obtained by calculating the states distribution of the individual service centers and then accumulating these distributions. From the similar derivations of message delay in a store-and-forward communication network [13], we can obtain the average message sojourn time as follows:

$$W = \frac{\sum_{i=1}^{k} \dfrac{a_i * r_i}{r_i - a_i}}{\sum_{i=1}^{N} a_{0i}}. \tag{4}$$

It is noted that for an SSE, the value of a_i varies and depends on the set of stations that are connected to it. Therefore, it is necessary to adjust the allocation to find an optimal topology for this network.

4. TOPOLOGICAL OPTIMIZATION OF LINKED CLUSTER NETWORKS

In this section, we take the average message sojourn time as a performance measure of networks. Therefore, the problem of topological optimization can be described as follows:

Given.

(1) N stations (i.e. OWSs and OPs) and C SSEs;
(2) the EMA vector $(a_{01}, a_{02}, \ldots, a_{0N})$;
(3) the PIT matrix $[P_{i,j}]$

and

(4) the mean service rate r_i of the service centers.

Object. To minimize the average message sojourn time.

Over the design variable. Topology, i.e. the allocation of OWSs and OPs in each cluster.

Subject to the constraint. Every cluster contains one SSE and one OP.

4.1. Search algorithm

To evaluate the message sojourn time of a particular topology, we need to find the solution of a new queuing network. Since any OWS can be conceivably allocated to any cluster, there are a total of C^{N-C} possible topological designs. Except for those simple problems, exhaustive search to find an optimal solution may seem out of question. Therefore, we propose a heuristic algorithm to find a nearly optimal network topology. This algorithm requires at most $N - C$ calculations of queuing networks. Therefore, it reduces the computation time sharply and is useful in practical designs.

Basically, our problem shows some analogy with Bernard's study on the topological optimization problem. He proposes a heuristic algorithm to obtain both the optimal number of local networks to be built and the assignment of stations in each network [8]. However, there are some distinctions between his model and ours. For example, in our model, in order to balance each cluster's load, the OPs must be equally distributed among clusters. In addition, due to data distribution and other considerations, the number of clusters must be predetermined. Therefore, it is desirable to develop a new algorithm to solve our problem.

Our algorithm resembles Fox's marginal allocation which is very well-known [14]. In that allocation, at one time only one resource unit is allocated based on the evaluation of some test function. This incremental allocation continues until the cost exceeds some preassigned value. In our problem, since no cost limit is known beforehand, the termination of our algorithm is determined by comparing the performance of two successive allocations. If the succeeding allocation has better performance than the preceding allocation, then the algorithm continues, otherwise it terminates.

The following outlines the general idea behind the algorithm: first, a cluster, labeled $C0$, is constructed by putting all OWSs together with the OP that has the heaviest traffic with OWSs; all of the remaining OPs are equally distributed into other clusters, labeled Ci, $i = 1 \ldots C - 1$. Then we transfer one of the OWSs of the $C0$ cluster to a Ci cluster which has the largest message flow coming to or from the transferred OWS. The transferring process is repeated if the average message sojourn time continuously decreases.

Two questions may arise in the above algorithm. The first is: which OWS of $C0$ we should select to move? The second is: to which cluster should the selected OWS be destined? Before answering these two questions, we first define the following term (all of the following examples are based on the PIT matrix of Table 1):

Table 1. A PIT matrix example

	C0					C1	C2
	OWS1	OWS2	OWS3	OWS4	OP1	OP2	OP3
OWS1	0.0	0.1	0.2	0.1	0.4	0.1	0.1
OWS2	0.1	0.0	0.2	0.1	0.3	0.2	0.1
OWS3	0.2	0.1	0.0	0.2	0.2	0.3	0.0
OWS4	0.1	0.1	0.1	0.0	0.2	0.3	0.2
OP1	0.3	0.2	0.1	0.1	0.0	0.2	0.1
OP2	0.1	0.2	0.2	0.1	0.2	0.0	0.2
OP3	0.1	0.1	0.2	0.2	0.2	0.2	0.0

Definition 1. Local access proportion

Local access proportion of station i in cluster C_k (denoted as L^i_{Ck}) is the proportion of messages emitted by station i of cluster Ck to other stations in the same cluster. For example,

$$L^{\text{OWS1}}_{C0} = \frac{0.1 + 0.2 + 0.1 + 0.4}{0.1 + 0.2 + 0.1 + 0.4 + 0.1 + 0.1} = 0.8. \tag{5}$$

Definition 2. Remote access proportion

Remote access proportion of station i of cluster Ck to cluster Cl (denoted as $R^i_{Ck, Cl}$) is the proportion of messages emitted by station i in cluster Ck to cluster Cl. For example,

$$R^{\text{OWS1}}_{C0, C1} = \frac{0.1}{0.1 + 0.2 + 0.1 + 0.4 + 0.1 + 0.1} = 0.1. \tag{6}$$

Definition 3. Attraction

Attraction of cluster Cl for station i of cluster Ck (denoted as $A^i_{Ck, Cl}$) is the ratio of $R_{Ck, Cl}$ and L_{Ck}. For example,

$$A^{\text{OWS1}}_{C0, C1} = \frac{(6)}{(5)} = \frac{R^{\text{OWS1}}_{C}0, C1}{L^{\text{OWS1}}_{C0}} = \frac{1}{8}.$$

The larger the $A^i_{Ck, Cl}$ is, the higher the probability that a message will be transferred from station i in cluster Ck to cluster Cl.

Subsequently, the heuristic algorithm is described in detail as follows.

Step 1. Put all OWSs together with the OP that has the largest sum of $P_{i,j}$ in a cluster $C0$, where i indicates OWSs and j specifies this particular OP, the remaining OPs are equally distributed to other clusters, labeled C_i, $i = 1 \ldots C - 1$. For example, in Fig. 3(a), since $\Sigma P_{i, \text{op1}} > \Sigma P_{i, \text{op2}}$ and $\Sigma P_{i, \text{op1}} > \Sigma P_{i, \text{op3}}$, it can be deduced that OP1 and the four OWSs constitute cluster $C0$, also OP2 and OP3 constitute the cluster $C1$ and $C2$ individually.

Step 2. Initialize the message sojourn time W to infinite.

Step 3. For each service center, compute the message arrival rate (a_i) by equations (1) and (2).

Step 4. Is any $a_i \geqslant r_i$? If yes, then go to Step 8, otherwise continue.

Step 5. Compute the message sojourn time W' of this particular topology by equation (4).

Step 6. Calculate $\Delta W = W - W'$.

Step 7. If $\Delta W < 0$, then stop; otherwise continue.

Step 8. Change the system topology by transferring an OWS from the cluster $C0$ to one of the clusters Ci, $i = 1 \ldots C-1$. The transferring rule is to select a cluster Ci which has the largest attraction over one of the OWS of $C0$. Add this OWS to the selected cluster and then delete it from the $C0$. For example, in Fig. 3(b), as the maximum of $A^i_{C0,C1}$ (equal to $A^{OWS4}_{C0,C1}$) is greater than that of $A^i_{C0,C2}$, OWS4 is transferred from $C0$ to $C1$.

Return to Step 3 and continue.

4.2. *Empirical validation of the search algorithm*

Since we cannot guarantee that the search algorithm as described above will ever find an optimal topology, we have compared the solutions found by the heuristic search to the optimal topology found through exhaustive search. The topology found by random allocation is also compared to show the effectiveness of the heuristic search. To complete the exhaustive search in a reasonable time, we restricted the problem size to 3 clusters and 12 stations. This gives a total of 3^9 ($=19683$) queuing networks to be solved to determine an optimal topology.

To test the flexibility of our search algorithm, we constructed the PIT matrix by random number generation. The sum of each row in the matrix equals to one (see Table 2). The service rate vector for the service centers of this system is also shown in Table 2. By defining the error ratio of some topology as $(A - B)/B * 100\%$, where value $A - B$ is the difference of the average message sojourn time between that topology and the optimal topology and value B is the average message sojourn time of the optimal topology, we varied the exogenous message arrival rate to obtain the error ratio of the nearly optimal topology found by the heuristic search and the error ratio of the topology built by random allocation. Table 3 shows the results of this experiment.

From Table 3, we see that the maximum error ratio of the heuristic results is below 5%. On the other hand, the maximum error ratio of the randomly allocated topology may be up to 26%. In this experiment, we have also varied the contents of PIT matrix to represent the various inter-relationships of the network nodes. Our experiment showed that if the inter-relationships of the nodes is split into clusters, this heuristic algorithm is more effective than the random allocation

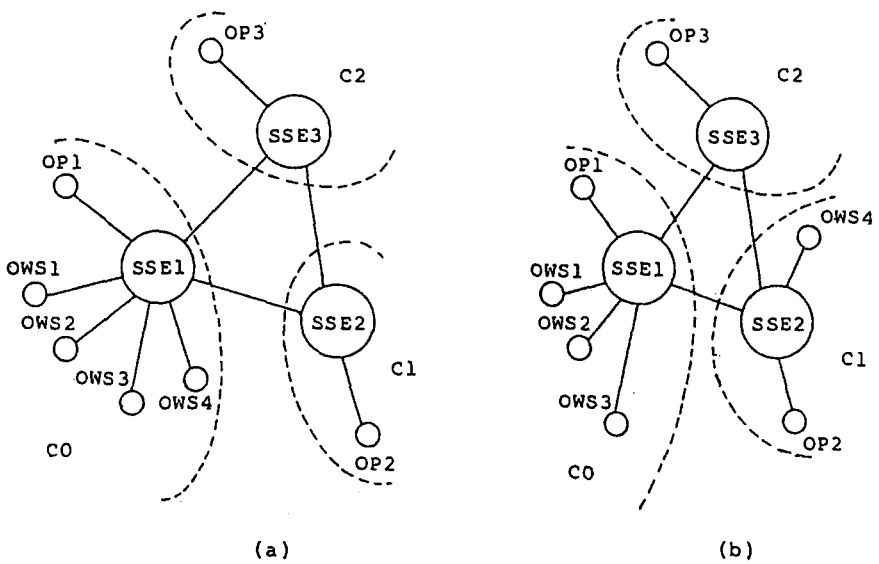

Fig. 3. Example of transferring an OWS. (a) Before transfer; (b) after transfer.

Table 2(a). Input parameters for the PIT matrix experiment

		OWS									OP		
		1	2	3	4	5	6	7	8	9	1	2	3
	1	0.000	0.110	0.134	0.185	0.051	0.044	0.059	0.044	0.020	0.186	0.109	0.059
	2	0.215	0.000	0.056	0.166	0.066	0.058	0.191	0.059	0.027	0.017	0.006	0.138
O	3	0.016	0.180	0.000	0.071	0.028	0.014	0.142	0.162	0.127	0.162	0.070	0.027
W	4	0.151	0.112	0.002	0.000	0.160	0.163	0.063	0.034	0.082	0.090	0.109	0.034
S	5	0.053	0.151	0.021	0.143	0.000	0.060	0.150	0.148	0.016	0.153	0.059	0.047
	6	0.058	0.009	0.178	0.125	0.120	0.000	0.159	0.146	0.008	0.137	0.028	0.031
	7	0.121	0.055	0.157	0.009	0.101	0.109	0.000	0.019	0.092	0.159	0.078	0.100
	8	0.025	0.057	0.067	0.149	0.095	0.150	0.149	0.000	0.127	0.051	0.094	0.037
	9	0.025	0.104	0.047	0.121	0.088	0.078	0.148	0.152	0.000	0.046	0.080	0.109
O	1	0.160	0.086	0.027	0.014	0.182	0.169	0.095	0.025	0.089	0.000	0.061	0.072
P	2	0.106	0.123	0.146	0.133	0.119	0.010	0.064	0.046	0.115	0.037	0.000	0.099
	3	0.013	0.035	0.071	0.133	0.087	0.162	0.134	0.120	0.047	0.025	0.173	0.000

Table 2(b). Input parameters for the service rate of the service centers experiment

Service center identifier	OWS									OP			SSE		
	1	2	3	4	5	6	7	8	9	1	2	3	1	2	3
Service rate	0.52	0.54	0.51	0.60	0.56	0.54	0.63	0.52	0.47	0.55	0.50	0.47	1.0	1.0	1.0

method. For the PIT matrix of Table 4, for example, the error ratio of the heuristic search is zero, while the maximum error ratio of the random allocation method may be up to 60%. We can see that the exhaustive search takes about 60 min of computation time (on the IBM PC/AT) while each heuristic search takes about 10 s of computation time. Thus, the heuristic algorithm appears to have been very successful and can be applied in the practical network design.

The relations of error ratio vs exogenous message arrival rate for both the heuristic search and the random allocation method is shown in Fig. 4. From this figure, we can see that as the message arrival rate increases, the error ratio of the random allocation increases sharply, while the error ratio of the heuristic results increase very slowly.

In short, this heuristic algorithm is especially applicable to the cases where the allocation of nodes is not easy to be determined by intuition. This heuristic algorithm could always find a better topology in contrast to the random allocation method.

5. CONCLUSIONS

In this paper, we have proposed an open queuing network to model a linked cluster network. Based on this model, we also present a heuristic algorithm to search for a near optimal network topology. Our experiment shows that the performance measure of the heuristic results is within 5% of the exhaustive search results. The heuristic search not only has much less computation time than the exhaustive search but also has a more accurate and stable performance measure than the random allocation method. Therefore, if the conventional network topological design methodologies also handle the queuing delays in the same way as we have presented in this paper, then the system performance would be evaluated more accurately.

Table 3. Experiment result

msg_arr_r	W in EXHST	W in HEUR	W in RNDALL	Error ratio HEUR(%)	Error ratio RNDALL (%)
0.08	8.86	9.05	11.22	2.1	26.6
0.07	8.16	8.27	9.16	1.4	12.3
0.06	7.57	7.64	8.07	0.8	4.0
0.05	7.08	7.12	7.34	0.5	3.6
0.04	6.58	6.58	6.78	0.0	3.0
0.03	6.05	6.05	6.34	0.0	4.6

EXHST—Exhaustive search; HEUR—heuristic search; RNDALL—random allocation.

Table 4(a). Another set of PIT matrix input parameters

		OWS									OP		
		1	2	3	4	5	6	7	8	9	1	2	3
O W S	1	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0
	2	0.2	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0
	3	0.2	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0
	4	0.0	0.0	0.0	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.6	0.0
	5	0.0	0.0	0.0	0.2	0.0	0.2	0.0	0.0	0.0	0.0	0.6	0.0
	6	0.0	0.0	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.0	0.6	0.0
	7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	0.0	0.2	0.6
	8	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.2	0.0	0.0	0.6
	9	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	0.0	0.0	0.0	0.6
O P	1	0.3	0.3	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	2	0.0	0.0	0.0	0.3	0.3	0.4	0.0	0.0	0.0	0.0	0.0	0.0
	3	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.3	0.4	0.0	0.0	0.0

Table 4(b). Another set of service rate of of the service centers input parameters

Service center identifier	OWS									OP			SSE		
	1	2	3	4	5	6	7	8	9	1	2	3	1	2	3
Service rate	0.34	0.34	0.36	0.34	0.34	0.36	0.34	0.34	0.36	0.56	0.56	0.56	1.0	1.0	1.0

Finally, we identify two other issues that deserve further investigation:

(1) In the current model, we assume that a station sends messages in a Poisson process with a fixed rate. In practice, a station does not send messages permanently but is active only during the time needed for sending m messages, after which the station enters a passive state for receiving messages only. For the QNM developed in Section 2, this amounts to the variation of an EMA vector. Therefore, one question worth addressing is: how to make this model more realistic by considering this variation?

(2) As all of the message passing in the system is handled by the SSEs (i.e. the cluster headers), the performance of the SSEs influences the overall system performance seriously. For the SSEs, the scheduling discipline affects the utilization of other service centers due to their correlations. This raises another problem: how to select the most appropriate scheduling discipline for the SSEs? which also appears to be an interesting topic.

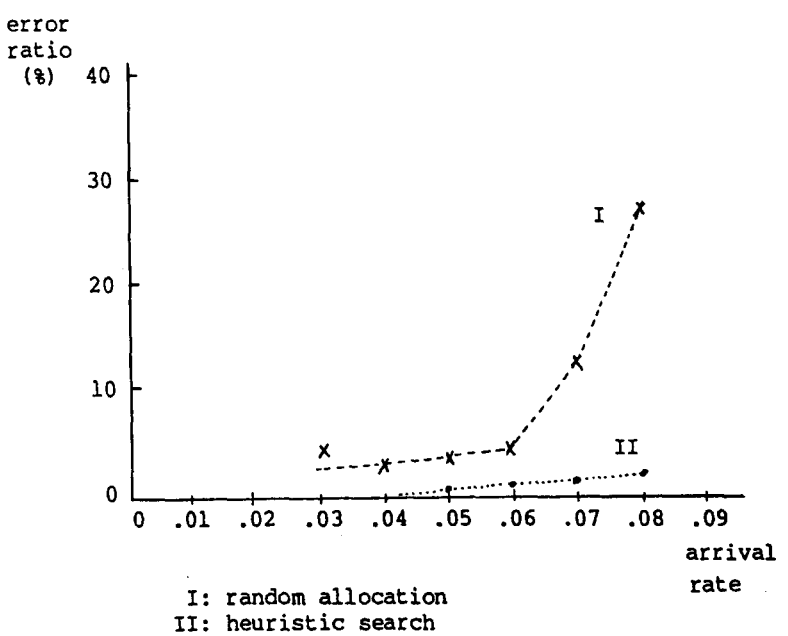

Fig. 4. Relationship of error ratio vs exogenous message arrival rate.

REFERENCES

1. R. R. Boorstyn and H. Frank, Large-scale network topological design. *IEEE Trans. Communs* **COM-25**(1), 29–47 (1977).
2. J. Martin, *Design and Strategy for Distributed Data Processing*. Prentice-Hall, Englewood Cliffs, N.J. (1981).
3. A. S. Tanenbaum, *Computer Networks*, pp. 32–88. Prentice-Hall, Englewood Cliffs, N.J. (1981).
4. M. Gerla and L. Kleinrock, On the topological design of distributed computer networks. *IEEE Trans. Communs* **COM-25**(1), 48–60 (1977).
5. P. Chen and J. Akoka, Optimal design of distributed information systems. *IEEE Trans. Communs* **COM-29**(12), 1068–1080 (1980).
6. Y. M. Chen, L. M. Tseng and S. M. Wang, The queuing model and topological optimization of office automation systems. *Proc. ICS'86*, Tainan, Taiwan, R.O.C., pp. 775–782 (1986).
7. C. M. Woodside and S. K. Tripathi, Optimal allocation of file servers in a local area network environment. *IEEE Trans. Softw. Engng* **SE-12**(8), 844–848 (1986).
8. G. Bernard, Interconnection of local computer networks: modeling and optimization problems. *IEEE Trans. Softw. Engng* **SE-9**(4), 463–470 (1983).
9. C. H. Sauer and K. M. Chandy, *Computer System Performance Modeling*. Prentice-Hall, Englewood Cliffs, N.J. (1981).
10. E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*, pp. 70–109, Academic Press, London (1980).
11. S. M. Wang, Building an office coordination supporting environment with message switching network. Master Thesis, National Cheng Kung University, Tainan, Taiwan, R.O.C., June (1986).
12. E. Cinlar, *Introduction to Stochastic Process* pp. 75–105. Prentice-Hall, Englewood Cliffs, N.J. (1975).
13. J. F. Hayes, *Modeling and Analysis of Computer Communications Networks*, Plenum Press, New York (1984).
14. B. Fox, Discrete optimization via marginal analysis. *Mgmt Sci.* **13**(3), 210–216 (1966).